2D Arrays Introduction

Easy

1

Are They Soulmates?

Medium

2

Digital Biometrics

Easy

3

Digital Footprint

Easy

4

Your classroom has 3 rows and 4 columns of seats. The seating arrangement is stored in a 2-D list:

```
seats = [["Alice", "Bob", "Carol", "David"], ["Eve", "Frank", "Grace", "Heidi"], ["Ivan", "Judy", "Mallory", "Niaj"]]
```

On python:

1. Print the name of the student sitting in the 2nd row, 3rd column
2. Print all students in the first row
3. Print the entire chart, one row per line
4. Ask the user for a name and tell whether the student is present or absent

Task: given two names as inputs, determine if their relationship is “magical.”

A magical bond exists if the string formed by combining both names reads the same forwards and backwards

Example:
input_1: “ana”; input_2: nana
output: “magical!”

here, “ana”+”nana” = “ananana” is a palindrome!

Think about the following questions and write a piece answering them all and sharing your thoughts.

1. What is biometric data? What forms of it do you use every day without realizing?
2. Where is your data going — stored locally on device, or uploaded to a server?
3. What’s the biggest risk if biometric data is leaked? (Hint: can you change your face?)

Look into one real-world incident of data leakage (e.g., Aadhaar biometric leak, Clearview AI, etc.)

Look up what digital footprints are.

Do a backtracking and identify where all you left digital footprints today. Think about how a digital footprint works.

You search something on amazon and the ad for the same product appears on instagram or youtube, how? What role does digital footprint play?

Explore digital forensics and see how digital footprints are used.

History of Encryption

Easy

5

Invariants

Hard

6

Nested Loops

Easy

7

Puzzle Heist

Hard

8

Encryption goes way back in history. Much before computers were invented.

Read up about the history of encryption and cryptography and the various ways of encrypting messages in different periods of time.

How good or bad were they as compared to what we have today?

Look up what invariants are.

Write the invariants of the algorithms we are writing in the class.

You can find the algorithms from the notebooks or challenge cards.

```
1
1 2
1 2 3
1 2 3 4
```

Write a python code that prints out the above pattern.

Tip: think of how having a loop inside another loop can achieve this.

Doing this challenge requires you to have finished the class activity: Puzzle Heist.

Write a python program for the entire puzzle that outputs the final 4-digit number.

RSA Encryption

Hard

9

Terms and Conditions

Easy

10

The Cult of Pattern

Hard

11

Caesar Ciphers

Easy

12

Task A:
Write an algorithm that performs the RSA encryption of a message. Clearly state the invariants. (How should one go about decrpyting the message?)

Task B:
Think of a way to break an RSA encryption. Write a python code that executes your logic and breaks a given RSA encrypted message. Clearly state the invariants. (What did you observe?)

Task C:
What makes the RSA algorithm so special?

For your favorite or most used website/app, read the T&C and Privacy Policies.

Identify what all data they are collecting about you, why they are interested in the data, and whether you should care about it.

Resource: https://tosdr.org/en

Check out the pattern shown below:

1. Use loops and recreate the pattern!
2. Refactor your code into 2 or more functions — because the pattern is symmetrical. Can you figure out the best way to break it up?

You've been hired to help a robot design a welcome banner using only letters and ASCII codes. Your task is to write a function that:

1. Takes a word as input. *("DOG")*
2. Converts each letter into its ASCII value using ord(). *("D" → 68)*
3. Shifts each ASCII value by +1 and prints the new character using chr(). *(69 → "E"; DOG → EPH)*

Can you recover the initial message?

Uniquely Decodable Code and Huffman Coding

Hard

13

Visual Cryptography

Easy

14

Visual Encodings

Easy

15

16

Sentence = Hey, by doing this activity, you just learned what uniquely decodable code is. Good job!

1. **Frequency Table**: count each character's frequency
2. **Huffman Tree**: use bottom-up approach to combine the two least frequency characters at each step; produce the final tree with chars as leaf nodes
3. **Assign Code**: assign binary to each character/leaf (left = 0; right = 1)
4. **Encode**: encode the full sentence into a binary string using the Huffman code you wrote.
5. **Decode**: given only the binary string, decode it back to the original sentence. Can you do it? Share your thoughts!

Go through what visual cryptography is and use the technique to encrypt a message you want to share with others.

Ask you friends to decrypt the message you encrypted!

Resource:
https://www.101computing.net/visual-cryptography/

Look up what Steganography is and go through the site mentioned below.

Resource:
https://stylesuxx.github.io/steganography/

1. How is this different from encryption?
2. How does the information get encrypted in a file? Can we use any file format? Try with all that you can.
3. How does it get decrypted?
4. What does the binary representation of a message mean? Why are you provided with it?